

Inteligência Artificial

Projeto I • Relatório
Grupo 44 • Mariana Mendes, 83521 • Miguel Regouga, 83530
Ano Letivo 2017/2018

Procura em Profundidade Primeiro (DFS)

	Tempo de execução (s)	Nº de nós expandidos	Nº de nós gerados
4x5 (2 cores, s/ solução)	0,061	1	0
4x5 (3 cores, c/ solução)	0,064	4	7
10x4 (3 cores, s/ solução)	0,062	74.702	74.701
10x4 (3 cores, c/ solução)	0,059	54	85
10x4 (5 cores, c/ solução)	0,063	3.123.308	3.123.363

Procura Gananciosa (Greedy)

	Tempo de execução (s)	Nº de nós expandidos	Nº de nós gerados
4x5 (2 cores, s/ solução)	0,059	1	0
4x5 (3 cores, c/ solução)	0,066	3	6
10x4 (3 cores, s/ solução)	0,067	74.702	74.701
10x4 (3 cores, c/ solução)	0,061	42	59
10x4 (5 cores, c/ solução)	0,063	4.637	4.692

Procura em A* (A-Star)

	Tempo de execução (s)	Nº de nós expandidos	Nº de nós gerados
4x5 (2 cores, s/ solução)	0,061	1	0
4x5 (3 cores, c/ solução)	0,063	3	6
10x4 (3 cores, s/ solução)	0,060	74.702	74.701
10x4 (3 cores, c/ solução)	0,062	42	59
10x4 (5 cores, c/ solução)	0,065	1.500	1.645

Análise dos Resultados

Para a obtenção dos resultados acima, o nosso grupo utilizou o exemplo que se encontrava no final do ficheiro “search.py”. A execução desta função demorou alguns minutos, sendo o resultado final guardado num ficheiro de *output* que indica o número de nós gerados e expandidos em cada tabuleiro e em cada procura (em profundidade primeiro, gananciosa e A*). Para cada input, usámos o comando “time python3 proj.py < input”, de modo a obter e registar o tempo de execução do nosso programa em cada caso.

Numa primeira análise, podemos verificar que, em qualquer procura, quanto maior é o número de cores, mais nós são gerados e expandidos.

O facto de um tabuleiro ter ou não ter solução influencia, naturalmente, a quantidade de nós expandidos e gerados: caso não haja solução, este número é bastante superior em comparação a um caso com solução,

A dimensão do tabuleiro tem também um impacto no número de nós expandidos e gerados — quanto maior o tamanho do mesmo, mais nós serão gerados e expandidos.

Analisando individualmente os resultados de cada procura, podemos verificar que a procura em profundidade primeiro é, de longe, a mais ineficiente. Num tabuleiro com dimensão 10x4, com 5 cores e com solução, são gerados mais de três milhões de nós — em comparação, a procura gananciosa e a procura em A* geram, respetivamente, 4.639 e 1.502, uma diferença bastante grande.

Assim, podemos concluir que a procura em A* é a mais eficiente das três para o cálculo de uma solução para o jogo Same Game, pois o número de nós expandidos e gerados é inferior às outras duas procuras.